REMARKS

In view of the above amendments and the following remarks, reconsideration of the objection and rejections and further examination are respectfully requested.

Claims 1-11 were rejected under 35 U.S.C. § 112, second paragraph for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Specifically, the rejection states that the claims are generally narrative and indefinite and that they appear to be a literal translation into English from a foreign document. This rejection is respectfully traversed, and it is respectfully submitted that claims 1-11 do point out and distinctly claim the subject matter recited therein. However, please note that independent claim 1 has been amended to clarify the features recited therein and improve readability. Further, please note that although the claimed invention includes several complex features of a processing apparatus, it is respectfully submitted that the features of the invention are distinctly claimed. Thus, in view of the above, this rejection is no longer applicable to claims 1-11.

Claims 1-11 were rejected under 35 U.S.C. § 112, first paragraph for failing to comply with the enablement requirement. Specifically, the Examiner is apparently taking the position that the "empty data" feature recited in the claims is not properly enabled by the original disclosure. This rejection is respectfully traversed for the following reasons.

The Applicants would like to thank the Examiner for briefly discussing this issue via telephone on September 4, 2007. During the telephone conversation, the term "empty data" was discussed, and the Examiner suggested that the Applicants explain the same in the response. In view of the above, it is noted that claim 1 recites that the empty data storage section receives data, erases the received data, and outputs "empty data" (i.e., 0 byes, null data) in response to a read request from another device.

It is well known in the art of data processing that empty data can be transmitted in response to a read request (see paragraph [0028] "specific example of the empty data storage section 23 is a null device which is supported by some OS's.") in circumstances when a read request is made by a device but the device should not actually receive data in response to the read request. There are many uses in the art of data processing for the above-mentioned feature, such as, for example, allowing the device which makes a read request to continue under normal operation (i.e., not stopping) even when the device should not be receiving new data. Moreover, it is noted that a device can be programmed to respond in a certain way when "empty data" is

received in response to a read request, thus making the received empty data an important feature of the operation of the device.

In addition, it is also noted that Paul Sheer's "Rute User's Tutorial and Exposition" (Sheer), cited by the Examiner, also teaches this above-mentioned "empty data" (null data) concept. For example, Sheer states "that dev/zero (1) Produces zero bytes, and as many of them [zero bytes] as you need. This is useful if you need to generate a block of zeros for some reason. Use dd (see Section 18.5.2) to read a specific number of zeros. /dev/null (1) Null device. Reads nothing. Anything you write to the device is discarded. This is very useful for discarding output." Thus, it is clear that Sheer teaches producing "empty data" (zero bytes) and reading "empty data" (zero bytes read by a null device).

Accordingly, in view of the above, it is clear that a person of ordinary skill in the art of data processing would be enabled to make and/or use the claimed invention. Thus, for the reasons discussed above, this rejection is not applicable to claims 1-11.

Page 4 of the Office Action states that a "substitute specification in proper idiomatic English and in compliance with 37 CFR 1.52(a) and (b) is required. The substitute specification filed must be accompanied by a statement that it contains no new matter." It is respectfully submitted that a substitute specification and abstract were filed with the Amendment of June 6, 2007. Further, it is submitted that the second paragraph on page 8 of the Amendment of June 6, 2007 states that "no new matter has been added" to the substitute specification and abstract.

Claims 1, 4-7, 9, and 11 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Parry et al. (U.S. 6,748,481 B1) in view of Sheer. Further, claims 2-3, 8 and 10 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Parry and Sheer and further in view of Barton et al. (U.S. 6,233,389 B1). The above-mentioned 35 U.S.C. § 103(a) rejections are clearly inapplicable to claims 1-11 for the following reasons.

Amended independent claim 1 recites a stream data processing apparatus for performing multiple processing steps during a processing of stream data. The stream data processing apparatus of claim 1 includes two processing sections (a transmitting-end processing section, and a receiving-end processing section), a control section, a temporary storage section, an empty storage section, and a connection management section. The two processing sections transmit/receive data to/from the temporary storage section and/or the empty storage section in

various combinations based on a change signal transmitted from a control section, clear request signals, and a state of the connection management section.

Specifically, claim 1 recites that (1) the control section transmits a change signal to both processing section, and that the change signal instructs a change of a subject of processing of the processing sections. Further, claim 1 recites that (2) the transmitting-end processing section performs a processing step and transmits the processed data, and the receiving-end processing section performs another processing step, wherein (3) the connection management section switches a write destination of the processed data transmitted from the transmitting-end processing section (switching between writing the processed data to the temporary storage section and the empty storage section) and switches a read source of data received by the receiving-end processing section (switching between reading data from the temporary storage section and the empty storage section). The Parry reference fails to disclose or suggest the above-discussed features (1)-(3) of claim 1.

Rather, Parry teaches a streaming information including a buffer IO layer 200, including a writer module 122, which receives blocks of streaming information and writes the blocks to the buffer IO layer 200, and including a read module 126 which selectively reads the blocks from the buffer IO layer 200 (see abstract; Fig. 7; col. 8, lines 54-57 and 64-67). In addition, Parry teaches the writer 122 can be blocked from writing to the buffer IO layer 200 and the reader 126 can be blocked from reading from the buffer IO layer 200 (see col. 13, lines 15-33). Further, Parry teaches that this "blocking," is controlled directly by both the writer 122 and the reader 126 (see Figs. 9A and 9B steps 301-310 and steps 311-317; and Figs. 11A and 11B).

Thus, it is clear that Parry teaches that the writer and the reader control the blocking of data to be written/read to/from the buffer IO layer, but does not disclose or suggest the connection management section which switches the write destination between the temporary storage section and the empty storage section and which switches the read source (of the receiving-end processing section) between the temporary storage section and the empty storage section, as recited in claim 1. In other words, Parry teaches that the writer and the reader themselves control/block the reading and the writing to/from the buffer IO layer, but does not disclose or suggest a connection management section which controls the reading and the writing by switching the write destination and the read source, as required by claim 1.

Further, in view of the above, it is clear that Parry does not disclose or suggest that the receiving-end processing section and/or the transmitting-end processing section receives a change signal from the control section, wherein the change signal instructs a change of the subject of processing (i.e., performing another processing step), because Parry merely teaches that the writer and the reader themselves control/block the reading/writing to/from the buffer IO layer. Therefore, it is clear that the Parry reference does not disclose or suggest the above-mentioned features of claim 1.

Further, claim 1 recites that (1) the transmitting-end processing section outputs a transmitting-end clear request to the connection management section if a change signal is received from a control section and (2) the receiving-end processing section outputs a receiving-end clear request to the connection management section if a change signal is received from the control section, wherein (3) the state of the connection management section is determined according to the receipt and order of receipt of both of the clear requests, and wherein the state of the management section controls the switching of the write destination and the read source. The Parry reference fails to disclose or suggest the above-mentioned features (1)-(3) as recited in claim 1.

Rather, Parry teaches that the above-mentioned "blocking" by the reader/writer is accomplished according to a synchronization algorithm of the buffer IO layer 200, wherein the blocking by the reader occurs after the reader detects a state of the writer and the blocking by the writer occurs after the writer detects a state of the reader (see col. 13, lines 15-33).

Thus, in view of the above, it is clear that Parry teaches that the "blocking" occurs only when the reader detects a state of the writer and/or the writer detects a state of the reader, but does not disclose or suggest that the clear request transmitted by the two processors determines the state of the connection management section based on the receipt and order of receipt of both of the clear requests, which in turn controls the switching of the write destination and the read source, as recited in claim 1. In other words, Parry teaches that the "blocking" occurs based on the reader/writer detecting the state of the writer/reader respectively, but does not disclose or suggest that the two processors output clear requests which determine the state of the connection management section, which in turn controls the above-mentioned switching.

Further, in view of the above, it is clear that based on the above-described configuration the transmitting-end processing section and the receiving-end processing section can operate

without detecting the state of each other (i.e., transmitting-end detecting state of receiving-end, and vice-versa). This feature of claim 1 allows a configuration of the two processing sections to be simplified. The invention of Parry does not include such a benefit, since the reader/writer are required to detect the operating state of each other in order to execute the "blocking." Therefore, it is clear that the Parry reference does not disclose or suggest the above-mentioned features of claim 1.

On page 5 of the Office Action the Examiner equates the connection management section of claim 1 with the buffer IO layer 200 of Parry. The Examiner's position is respectfully traversed for the reasons discussed below.

As mentioned above, claim 1 recites that the connection management section controls the write destination of the transmitting-end processing section (i.e., the write destination is switched between the temporary storage section and the empty storage section) and controls the read source of the receiving-end processing section (i.e., the read source is switched between the temporary storage section and the empty storage section). Rather, Parry teaches that the buffer IO layer 200 merely interfaces with the reader/writer but does not control the read source and the write destination.

This feature of claim 1 allows the processing of the two processing sections to be continuous and uninterrupted since the processing section can continue to process data and the connection management section can control the write destination and the read source (i.e., the processors themselves can process data without controlling the read source and/or write destination). The invention of Parry does not include such a benefit, since the reader/writer do not continue to operate, but are blocked from reading/writing from the buffer IO layer.

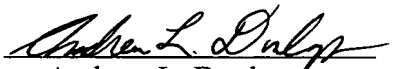
Thus, it is clear that Parry does not disclose or suggest the connection management section which controls the reading/writing to/from the temporary storage section and the empty storage section.

In view of the above, it is respectfully submitted that the Parry reference does not anticipate the invention as recited in amended claim 1. Furthermore, the Sheer and Barton references do not disclose or suggest any of the above-discussed features of independent claim 1 which are lacking from the Parry reference. As a result, the above-mentioned references, individually or collectively, fail to render obvious amended independent claim 1 or the claims that depend therefrom.

In view of the above amendments and remarks, it is submitted that the present application is now in condition for allowance and an early notification thereof is earnestly requested. The Examiner is invited to contact the undersigned by telephone to resolve any remaining issues.

Respectfully submitted,

Satoshi INAMI et al.

By: 

Andrew L. Dunlap
Registration No. 60,554
Attorney for Applicants

WDH/ALD/nrj
Washington, D.C. 20006-1021
Telephone (202) 721-8200
Facsimile (202) 721-8250
November 16, 2007